In the Specification

Please amend the specification in accordance with the marked-up paragraphs below.

The paragraph beginning on page 2 line 24 is amended to read:

Pseudo-random scan testing typically involves placing the circuit in a test mode to form the one or more scan chains referred to above. Then, the scan chain is injected with a pseudo-random test pattern, and the circuit temporarily returned to its normal (non-test) state to allow it to execute at least one normal cycle. The circuit is then returned to the test mode, and the resultant state extracted and combined with other extracted states to form a "test signature." The test signature is compared to a "golden" signature developed, e.g., from running a simulation of good circuit. The compare provides the GO/NO-GO indication of the operability of the circuit. Examples of this technique can be seen in U.S. Patent Nos. 4,718,065, 5,694,452, and 6,029,263.

The paragraph beginning on page 3 line 18 is amended to read:

An integrated circuit typically includes ~~of~~ a number of state machines, each having a number of flops, and each forming a clock domain that is normally asynchronously operated relative to the other click domains. In order to be able to test the entire integrated circuit at once (i.e., without resorting to sequentially testing the circuits within each clock domain), appropriate timing must be established at the interfaces between the clock domains to ensure deterministic operation for testing. Each clock domain will have its own clock distribution network, which meets functional insertion delay and skew requirements according to the application. Skew, in this context, is the difference in arrival times at clocked circuits of what is logically a single clock edge. For example, referring to Fig. 1, there is illustrated a representative integrated circuit, designated generally with the reference numeral 10, with three distinct clock domains 1, 2, and 3. Each clock domain includes at least one edge-triggered flop. Thus, clock domain 1 has at least the flop FF1, clock domain 2 includes at least the flops FF2, FF21, FF22, and clock domain 3 includes flop at least FF3. Each clock domain (1, 2, and 3) may also include combinatorial logic C (C1, C2, and C3, receptively). Data is received by each of the clock

domains 1, 2, 3 at PORT 1, PORT 2, and PORT 3 inputs, respectively, while separate and different (and not necessarily synchronous) clocking signals are received at the CLOCK1, CLOCK 2, and CLOCK 3 inputs. Clock domain 2 (i.e., flop FF2) receives, as inputs on signal lines 12 and 14, data outputs from flops FF1 and FF3 of clock domains 1 and 3, respectively, as well as self-synchronous inputs. Flops FF21 and FF22 are employed between the asynchronous domains to bound the arrival time of signals from domains 1 and 3 into domain 2. The logic design of domain 2 must incorporate provision for the metastable behavior of these flops, if the behavior is to be reliable.

The paragraph beginning on page 5 line 8 is amended to read:

Such scan testing requires that the circuitry of the individual clock domains 1,2,3, each of which will contain at least one state machine, operate as a single synchronous unit in order that the scan test results be predictable and repeatable. The functionally disjoint clock trees of each clock domain are fed from a common test clock source when in scan test mode. The interfaces between the clock domains, which were formerly asynchronous, may now have short-path problems which will cause unreliable/unrepeatable scan test results. This problem arises because, as is conventional, edge-triggered devices will accept and latch data applied to the input on one edge of the applied clock. Thus, it is not entirely certain, for example, whether the prior output of flop e.g. FF1 or the new output is transferred to flop e.g. FF21. The result depends on the order in which Clock 1 is received at FF1 relative to when Clock 2 is received at FF21. This order depends on the summation of delays from a common timing reference point somewhere in the tester to the respective flops. Some of the summation terms (hence the clock arrival times) are not knowable a priori. This condition is not indicative of unreliable functional operation, and is relatively easy to avoid in latch-based designs such as that used by IBM, which tend to avoid a lot of these problems by using a latch-based, two-phase, non-overlapped clock design discipline. A description of a latch-based system may be found ~~at~~ in an IBM ASIC Product Applications Note entitled "ASIC Design Methodology Primer", document number SA14-2314-00, 1998.

~~http://www.chips.ibm.com/products/asics/document/appnote/231400_0.pdf.~~

The paragraph beginning on page 7 line 3 is amended to read:

Accordingly, the digital logic is subjected to conventional scan testing when the scan test signal is asserted, except that the scan test signal is accompanied by a scan clock that is applied to the latches added according to the present invention. A test vector is scanned into the digital logic, and the scan test signal is de-asserted to allow the digital logic to resume its functional configuration. The test clock is then asserted to temporarily freeze (when in one state, e.g., a LOW state) the operational domain crossing interface signals and the logic is allowed to operate normally, for one or more clocks. For each assertion of the functional clocks, the test clock will likewise ~~by~~ be cycled. Then, the scan test signal is asserted to again form the scan chain, and the resultant state shifted from the digital logic while a new test vector may be shifted in.

The paragraph beginning on page 8 line 18 is amended to read:

Turning now to Fig. 3, there is illustrated the circuit 100, which is, in effect, a modified version of the circuit 10' of Fig. 2, incorporating an embodiment of the present invention. As Fig. 3 shows, the circuit 100 includes, as did circuit 100, three clock domains 102, 104, and 106. Each clock domain includes data (DATA) and clock (CLK) inputs, combinatoral logic C, and representative clocked, edge-triggered devices such as the flops FF1 (clock domain 102), F2, FF21, and FF22 (clock domain 104), and FF3 (clock domain 106). Multiplexers 120, 122, ..., 128 are provided in the circuit 100 to allow formation of a selectable scan path beween an SDI input on signal line 130 to a SDO output line 112b in response to assertion of a test signal (SCAN TEST (ST) on signal line 132.

The paragraph beginning on page 9 line 7 is amended to read:

Multiplexers 120, 124, ..., 128 provide the circuit 100 with data paths that route functional data from source (e.g., FF1) to destination (e.g., FF21) when circuit 100 is operating normally. Assertion of the SCAN TEST signal, however, operates the multiplexers 120, 122, ..., 128 to select a data path that forms a scan chain for scan testing operation so that scan data introduced to the digital system 100 at the SDI input on signal line 130 is removed from the circuit at Q output of the flop FF3 at the SDO output after application of sufficient clock pulses

at clock terminals CLK1, CLK2, and CLK3. The SCAN TEST signal received on signal line 132 is applied to the control (C) input of the multiplexers 120, ..., 128. When the SCAN TEST signal is in one state (e.g., LOW), each of the multiplexers 120, ..., 128 is switched to pass functional data. ~~When, however~~However, when the SCAN TEST signal is asserted (e.g., HIGH), the multiplexers 120, ..., 128 are put in a state to pass scan data and connect the flops in one or more scan chains (i.e., extended shift register structures).

The paragraph beginning on page 8 line 29 is amended to read:

Note that there is ~~not~~ no operational data path from the clock domain 104 to the clock domain 106. Although the clock domain 106 could be tested separately from the clock domains 102, 104, and even in parallel, this would require additional test inputs (to bring in test data and clock signals) to be added. Thus, for scan tests, a data path 114b is added in order to include the circuitry of the clock domain 106 in the scan testing of the circuitry of clock domains 102, 104. Again, due to the problems encountered in passing data between different clock domains, a latch LT2 is added, clocked/enabled by TEST_CLK, and connected to receive the data output of the flop FF22 to pass it to the flop FF3 (via the multiplexer 128).

The paragraph beginning on page 10 line 20 is amended to read:

To better understand the operation of the latches LT1, .., LT3, Figs 4 and 5 are provided. Figs. 4A and 4B show the flops FF1 and FF21 and an associated timing diagram, respectively, without the latch LT1. Figs 5A and 5B illustrate operation of the flops with the addition of the latch LT1. Referring first to Fig. 4A, shown is the data path formed between the flops that includes the multiplexer 120 – as shown also in Fig. 3. Assume that scan testing is initiated, and the clock signals application to the clock inputs (C) of the flops FF1 and F21 are an early clock (E_CLK) and a late clock (L_CLK) so called because the rising or clocking edge of the E_CLK occurs before the clocking or rising edge of the L-CLK. (It will be evident to those skilled in this art that the flops FF1 and FF21, as well as the rest of the flops of circuit 100 of Fig. 3 could be clocked or triggered by the falling edge of the clock signal applied to the C inputs of each.)

The paragraph beginning on page 10 line 21 is amended to read:

~~Fig. 5 illustrates~~Figs. 5A and 5B illustrate the solution of the present invention by the addition of the latch LT1. As has been indicated, latch LT1 is of a type that is transparent when one state (e.g., when high) of a clock signal is applied to the clock input (CK), but that captures and holds data by the transition to the other of the two states ~~(e.g., high to low)~~that the clock signal can assume (e.g., high to low). Thus, referring to Figs. 5A and 5B, now when the flop FF1 is clocked by the rising edge 140' of the E_CLK, changing the output Q of the flop to the data 152', the flop FF21 does not see the new data; it is blocked by the latch LT1 being held in the capture state by the low state of the TEST_CLK signal. When the flop FF21 is clocked by the rising edge 146' of the L_CLK, it will capture and hold the data supplied by the output Q of the latch LT1 – which was the data 150' at the output of the flop FF1 prior to the edge 140'.

The paragraph beginning on page 10 line 32 is amended to read:

Then, when the TEST_CLK signal changes from its low state 160 to its high state 162, the output data of the flop FF1 will be communicated to the flop FF21. That data is captured and held when the TEST_CLK drops again to its low state 160 so that the next transitions of the E-CLK and L_CLK will not have the flops FF1 and FF21 possibly capturing and holding the same data. Rather, the flop FF1 captures new data 154 while the flop FF21 captures and holds the data held by the latch LT1, i.e., the data 152' that was previously held~~help~~ by the flop FF1 before the last change. Thus, it can be seen that the transfer of scan data across the clock domains is accomplished in an ordered fashion so that the scan testing is not affected by timing differences between the domains.